Claim Amendments

1. (currently amended) An apparatus for scheduling the transmission of data packets for a plurality of data packet flows, said data packet flows being allocated ~~given~~ service shares of the transmission capacity $r$ of a communication link and being grouped in bundles, said bundles being allocated service shares $\underline{R_I}$ of the ~~processing~~ transmission capacity of said communication link, the transmission over the communication link being divided in service frames, a service frame offering at least one transmission opportunity to every data packet flow that is backlogged, a backlogged data packet flow being a data packet flow that has at least one data packet stored in respective one of a plurality of packet queues, the scheduling apparatus comprising:

means for determining the duration of the service frame; and

means for guaranteeing that each data packet flow always receives at least its allocated service share during a service frame if it remains continuously backlogged over the whole duration of said service frame, and that each bundle receives at least its allocated service share during a service frame if there is always at least one data packet flow in the bundle that remains continuously backlogged for the whole duration of said service frame, said guaranteeing means including:

means for maintaining, for each bundle $I$, a cumulative share $\Phi_I$ that relates to the sum of said service shares allocated to respective ones of said data packet flows that are grouped together in the same bundle $I$;

means for computing, for each bundle $I$, a service ratio between the service share $\cancel{R_I}$ allocated to said bundle $I$ and said cumulative share $\Phi_I$ of the bundle; and

X 1/5

means for modulating said service shares allocated to respective ones of said plurality of data packet flows using the service ratio computed for respective ones of said plurality of bundles.

2. (original) The scheduling apparatus of claim 1, wherein a Weighted Round Robin (WRR) algorithm is used to schedule the transmission of data packets.

3. (original) The scheduling apparatus of claim 1, wherein a Deficit Round Robin (DRR) algorithm is used to schedule the transmission of data packets.

4. (original) The scheduling apparatus of claim 1, wherein a Surplus Round Robin (SRR) algorithm is used to schedule the transmission of data packets.

5. (original) The scheduling apparatus of claim 1, wherein the duration of said service frames is variable.

6. (original) The scheduling apparatus of claim 1, wherein the duration of said service frames is fixed.

7. (original) The scheduling apparatus of claim 1, wherein said means for determining the duration of a service frame include:

a global frame counter *FRMCNT*;

a start flag $\sigma_I$ for each bundle *I* of said plurality of bundles; and

a frame flag $FF_i$ for each data packet flow $i$ of said plurality of data packet flows.

1

8. (original) The scheduling apparatus of claim 7, wherein the start flag $\sigma_I$ of bundle $I$ is set equal to the global frame counter *FRMCNT* when the first data packet flow in the bundle becomes backlogged.

1

9. (original) The scheduling apparatus of claim 7, wherein the frame flag $FF_i$ of data packet flow $i$ is set to a different value than the global frame counter *FRMCNT* when the flow becomes backlogged or is processed for the last time in the current service frame.

1

10. (original) The scheduling apparatus of claim 7, wherein the end of a service frame and the start of the following one are simultaneously detected when the frame flag $FF_i$ of the next data packet flow $i$ to be processed has different value than the global frame counter *FRMCNT*.

1

11. (original) The scheduling apparatus of claim 10, wherein the value of said global frame counter *FRMCNT* is set equal to the value of said frame flag $FF_i$ after detecting a difference between the two values.

1

12. (original) The scheduling apparatus of claim 1, wherein the value of the cumulative share $\Phi_I$ of bundle $I$ is equal to the sum of the service shares of the data packet flows of bundle $I$ that are backlogged.

1

13. (original) The scheduling apparatus of claim 12, wherein the value of the cumulative share $\Phi_I$ of bundle $I$ is set when a first data packet flow of the bundle is first serviced in a service frame, and kept unchanged for the whole

duration of the same service frame, even if the backlog state of one or a plurality of data packet flows of bundle $I$ changes during the service frame.

14. (original) The scheduling apparatus of claim 13, wherein a running share $\phi_I$ maintains the sum of the service shares of the data packet flows that are backlogged in bundle $I$, and changes when the backlog state of one or a plurality of data packet flows in the bundle changes, the value of said running share $\phi_I$ being used to set the value of said cumulative share $\Phi_I$ when required.

15. (original) The scheduling apparatus of claim 13, wherein said first service to said first data packet flow of bundle $I$ is detected when the start flag $\sigma_I$ of the bundle $I$ that includes the next flow $i$ to be processed has different value than the global frame counter *FRMCNT*.

16. (currently amended) A method for scheduling the transmission of data packets for a plurality of data packet flows, said data packet flows being allocated service ~~given~~ shares of the transmission capacity of an outgoing communication link and being grouped in a plurality of bundles, said bundles being allocated service shares $\underline{R_I}$ of the transmission capacity $r$ of said outgoing communication link, the transmission over the communication link being divided in service frames, a service frame offering at least one transmission opportunity to every data packet flow that is backlogged, a backlogged data packet flow being a data packet flow that has at least one data packet stored in respective one of a plurality of packet queues, the method comprising the steps of:

determining the duration of the service frame;

guaranteeing that each data packet flow always receives at least its allocated service share during a service frame if it remains continuously

backlogged over the whole duration of said service frame, and that each bundle receives at least its allocated service share during a service frame if there is always at least one data packet flow in the bundle that remains continuously backlogged for the whole duration of said service frame;

maintaining, for each bundle $I$, a cumulative share $\Phi_I$ that relates to the sum of said service shares allocated to respective ones of said data packet flows that are grouped together in the same bundle $I$;

computing, for each bundle $I$, a service ratio between the service share $\cancel{R_I}$ allocated to said bundle $I$ and said cumulative share $\Phi_I$ of the bundle; and

modulating said service shares allocated to respective ones of said plurality of data packet flows using the service ratio computed for respective ones of said plurality of bundles.